Lecture Note 20

1 Policy Search Methods

So far, we have focused on finding an optimal or good policy indirectly, by solving Bellman's equation either exactly or approximately. In this lecture, we will consider algorithms that search for a good policy directly.

We will focus on average-cost problems. Recall that one approach to finding an average-cost optimal policy is to solve Bellman's equation

$$\lambda e + h = Th.$$

Under certain technical conditions, ensuring that the optimal average cost is the same regardless of the initial state in the system, it can be shown that Bellman's equation has a solution (λ^*, h^*), corresponding to the optimal average cost, and h^* is the *differential cost function*, from which an optimal policy can be derived. An alternative to solving Bellman's equation is to consider searching over the space of policies directly, i.e., solving the problem

$$\min_{u \in U} \lambda(u), \tag{1}$$

where $\lambda(u)$ is the average cost associated with policy u and U is the set of all admissible policies. In the past, we have been most focused on policies that are stationary and deterministic; in other words, if $\mathcal{S}$ is the state space and $\mathcal{A}$ is the action space (consider a common action space across states, for simplicity), we have considered the set of policies $u : \mathcal{S} \mapsto \mathcal{A}$, which prescribe an action $u(x)$ for each state x. Note that, if U is the set of all deterministic and stationary policies, we have $|U| = |\mathcal{A}|^{|\mathcal{S}|}$, so that problem (1) involves optimization over a finite and exponentially large set (in fact, $|U|$ grows exponentially in the size of the state space, or double-exponentially in the dimension of the state space!).

In order to make searching directly in the policy space tractable, we are going to consider restricting the set of policies U in (1). Specifically, we are going to let U be a set of *parameterized* policies:

$$U = \{u_\theta : \theta \in \Re^K\},$$

where each policy u_θ corresponds to a *randomized* and stationary policy, i.e., $u_\theta(x, a)$ gives the probability of taking action a given that the state is x. We let g_θ, P_θ and $\lambda(\theta)$ denote the stage costs and transition probability matrix associated with policy u_θ:

$$\begin{aligned} g_\theta(x) &= \sum_a g_a(x) u_\theta(x, a) \\ P_\theta(x, y) &= \sum_a P_a(x, y) u_\theta(x, a) \end{aligned}$$

Example 1 (Threshold Policies) Admission Control

Suppose that we have a total amount R of a certain resource and requests from various types i for amounts r_i, $i = 1, \ldots, n$. Once a request is accepted, it occupies the amount r_i of resource for a certain length of time, before freeing it again. The admission control problem is to decide, upon arrival of a new request, whether to accept it or not. Let R_t the current amount of available resources. A possible threshold policy

for this problem could compare R_t with a certain threshold θ_i and only accept a new request of type i if $R_t \geq \theta_i$.

American Options

Consider the problem of when to exercise the option to buy a certain stock at a prespecified price K. A possible threshold policy is to exercise the option (i.e., buy the stock) if the market price at time t, P_t, is larger that a threshold θ_t.

Once we restrict attention to the class of policies parameterized by a real vector, problem (1) becomes a standard nonlinear optimization problem:

$$\min_{\theta \in \Re^k} \lambda(\theta). \tag{2}$$

With appropriate smoothness conditions, we can find a local optimum of (2) by doing gradient descent:

$$\theta_{k+1} = \theta_k - \gamma_k \nabla \lambda(\theta_k). \tag{3}$$

In the next few lectures, we will show that biased and unbiased estimates of the gradient $\nabla\lambda(\theta)$ can be computed from system trajectories, giving rise to simulation-based gradient descent methods.

1.1 A Convenient Expression for $\nabla\lambda(\theta)$

We first introduce assumptions that ensure the existence and differentiability of $\lambda(\theta)$.

Assumption 1 *Let $\rho = \{P_\theta | \theta \in \Re^k\}$ and $\bar{\rho}$ be the closure of ρ. The Markov Chain associated with P is irreducible and there exists x^* that is recurrent for every P.*

Assumption 2 *$P_\theta(x, y)$ and $g_\theta(x)$ are bounded, twice differentiable, with bounded first and second derivatives.*

Lemma 1 *Under Assumption 1 and 2, for every policy θ there is a unique stationary distribution π_θ satisfying $\pi_\theta^T = \pi_\theta^T P_\theta$, $\pi_\theta^T e = 1$, and $\lambda(\theta) = \pi_\theta^T g_\theta$. Moreover, $\lambda(\theta)$ and π_θ are differentiable.*

In order to develop a simulation-based method for generating estimates of the gradient, we will show that $\nabla\lambda(\theta)$ can be written as the expected value of certain functions of pairs of states (x, y), where (x, y) is distributed according to the stationary distribution $\pi_\theta(x)P_\theta(x, y)$ of pairs of consecutive states.

First observe that

$$\nabla\lambda(\theta) = \nabla\pi_\theta^T g_\theta + \pi_\theta^T \nabla g_\theta \tag{4}$$

It is clear that the secibd term $\pi_\theta^T \nabla g_\theta$ can be estimated via simulation; in particular, we know that

$$\pi_\theta^T \nabla g_\theta = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \nabla g_\theta(x_t).$$

Hence if we run the system with policy θ, and generate a sufficiently long trajectory $x_1, x_2, \ldots, x_T$, we can set

$$\sum_x \pi_\theta(x) \nabla_\theta g_\theta(x) \approx \frac{1}{T} \sum_{t=1}^{T} \nabla_\theta g_\theta(x_t).$$

The key insight is that the first term in (4), $\nabla\pi_\theta^T g_\theta$, can also also be estimated via simulation. In order to show that, we start with the following theorem.

Theorem 1 (Amazing Fact 1!)

$$\nabla \pi_\theta^T g_\theta = \pi_\theta^T \nabla P_\theta h_\theta, \tag{5}$$

where h_θ is the differential cost function associated with policy θ.

Proof: We have

$$\lambda(\theta)e + h_\theta = g_\theta + P_\theta h_\theta.$$

Multiplying both sides by $\nabla \pi_\theta^T$, we get

$$\lambda(\theta)\nabla \pi_\theta^T e + \nabla \pi_\theta^T h_\theta = \nabla \pi_\theta^T g_\theta + \nabla \pi_\theta^T P_\theta h_\theta.$$

Since $\pi_\theta^T e = 1$, $\nabla_\theta \pi_\theta^T e = 0$, and

$$\nabla \pi_\theta^T h_\theta = \nabla \pi_\theta^T g_\theta + \nabla \pi_\theta^T P_\theta h_\theta. \tag{6}$$

Moreover, since $\pi_\theta^T P_\theta = \pi_\theta^T$, we have

$$\nabla \pi_\theta^T P_\theta + \pi_\theta^T \nabla P_\theta = \nabla \pi_\theta^T$$

and

$$\nabla \pi_\theta^T P_\theta h_\theta + \pi_\theta^T \nabla P_\theta h_\theta = \nabla \pi_\theta^T h_\theta \tag{7}$$

The theorem follows from (6) and (7). □

It is still not clear how to easily compute (5) from the system trajectory. Note that

$$\pi_\theta^T \nabla P_\theta h_\theta = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} (\sum_y \nabla P_\theta(x_t, y) h_\theta(y)),$$

which suggests averaging $f(x) = \sum_y \nabla P_\theta(x, y) h_\theta(y)$ over a system trajectory $x_0, x_1, \ldots, x_T$, however this gives rise to two difficulties: first, we must perform a summation over y in each step, which may involve a large number of operations; second, we do not know $h_\theta(y)$. We can get around the first difficulty by employing an artifact known as the likelihood ratio method. Indeed, let

$$L_\theta(x, y) = \frac{\nabla P_\theta(x, y)}{P_\theta(x, y)}.$$

We make the following assumption about L_θ:

Assumption 3 *There is $B < \infty$ such that $L_\theta(x, y) \leq B$ for all θ.*

Assumption 3 is true, e.g., if for each pair (x, y), we have $P_\theta(x, y) = 0, \forall\, \theta$, or $P_\theta(x, y) \geq \epsilon, \forall\, \theta$. More concretely, a sufficient condition is $u_\theta(x, a) \geq \epsilon$, $\forall a \in A_x$.

Under this condition, we can rewrite (5) as

$$\begin{aligned} \pi_\theta^T \nabla P_\theta h_\theta &= \sum_x \pi_\theta(x) \sum_y \nabla P_\theta(x, y) h_\theta(y) \\ &= \sum_x \sum_y \pi_\theta(x) P_\theta(x, y) L_\theta(x, y) h_\theta(y), \end{aligned}$$

and assuming that we can compute or estimate h_θ, we can estimate (5) from a trajectory x_0, x_1, x_T by considering

$$\pi_\theta^T \nabla P_\theta h_\theta \approx \frac{1}{T} \sum_{t=0}^{T-1} L_\theta(x_t, x_{t+1}) h_\theta(x_{t+1}).$$

Our last step will be to show that we can get unbiased estimates of h_θ by looking at cycles in the system trajectory between visits to the recurrent state x^*. This follows from the following observation, which was proved in Problem Set 2:

Theorem 2 ***Amazing Fact 2***
Let x^* be a recurrent state under policy θ. Let

$$T = \min\{t > 0 : x_t = x^*\}$$

Then

$$\begin{aligned} h_\theta(x) &= \mathrm{E}\left[\sum_{t=0}^{T-1} (g_\theta(x_t) - \lambda(\theta)) | x_0 = x\right] \\ h_\theta(x^*) &= 0 \end{aligned}$$

is a differential cost function for policy θ.

Putting together all of the pieces we have developed so far, we can consider the following algorithm. Let $x_0, x_1, \ldots$ be a system trajectory. Let t_m be the time of the mth visit to the recurrent state x^*. Based on the trajectory $x_{t_m}, x_{t_m+1}, \ldots, x_{t_{m+1}}$, compute

$$\hat{h_\theta}(x_n) = \sum_{t=n}^{t_{m+1}-1} \left[g_\theta(x_t) - \tilde{\lambda}_m\right], n = t_m + 1, \ldots, t_{m+1} - 1,$$

and

$$F_\theta(\tilde{\lambda}_m) = \sum_{n=t_m}^{t_{m+1}-1} \left[\hat{h}_\theta(x_n) h_\theta(x_{n-1}, x_n) + \nabla g_\theta(x_n)\right].$$

Then $F_\theta(\tilde{\lambda}_m)$ gives a biased estimate of $\nabla\lambda(\theta)$, where the bias is on the order of $O(|\lambda(\theta) - \tilde{\lambda}_m|)$:

Theorem 3

$$\mathrm{E}_\theta\left[F_m(\tilde{\lambda})\right] = \mathrm{E}_\theta(T)\nabla\lambda(\theta) + G(\theta)\left(\lambda(\theta) - \tilde{\lambda}\right)$$

where $G(\theta)$ is a bounded function.

We can update the policy by letting

$$\begin{aligned} \theta_{m+1} &= \theta_m - \gamma_m F_{\theta_m}(\tilde{\lambda}_m) \\ \tilde{\lambda}_{m+1} &= \tilde{\lambda}_m + \eta\gamma_m \sum_{n=t_m}^{t_{m+1}-1} \left[g_{\theta_m}(x_n) - \tilde{\lambda}_m\right] \end{aligned}$$

where $\eta > 0$.

Assumption 4 *$\sum_{m=1}^{\infty} \gamma_m = \infty$ and $\sum_{m=1}^{\infty} \gamma_m^2 < \infty$.*

Theorem 4 *Under Assumptions 1, 2, 3 and 4, we have*

$$\lim_{m\to\infty} \nabla\lambda(\theta_m) = 0 \quad w.p.\,1.$$